# What Drives the Price of Used Cars

Tauseef Bashir
June 1, 2022

This document summarizes our process, finding, and recommendations using a Machine Learning system from scratch, including building a predictive model. Here is a high-level process I followed:

- Data cleaning
- Exploratory Data Analysis (EDA)
- Building machine learning models: Linear Regression, Ridge Regression, Lasso, and KNN
- Comparison of the performance of the models
- Summary findings of the study with recommendations and next steps

## The Problem

Due to supply chain issues, increased consumer demand, and inflation, the prices of used cars have been rising. With the higher price, limited supply of new vehicles, and the inability of customers to buy new cars due to the lack of funds due to the current economic conditions, used car sales are on the rise globally. Therefore, there is a need for a used car price prediction system to effectively determine the worthiness of the car using a variety of features. The dealerships can use these models to plan and optimize their inventory and have better visibility into a used car's actual market value.

## The Client

Predicting used cars' market value can help buyers and sellers.
**Used car sellers (dealers):** They are one of the most prominent target groups that can be interested in the results of this study. If used car sellers better understand what makes a car desirable and what the essential features are for a used car, then they may consider this knowledge and offer a better service.

# The Data

The dealership network provided the data used in this project for analysis.

```
df=pd.read_csv('/content/drive/MyDrive/practical_application_II_starter (2)/data/vehicles.csv')
```

Let's take a look at the csv

```
df.head(10).transpose()
```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| **id** | 7222695916 | 7218891961 | 7221797935 | 7222270760 | 7210384030 | 7222379453 | 7221952215 | 7220195662 | 7209064557 | 7219485069 |
| **region** | prescott | fayetteville | florida keys | worcester / central MA | greensboro | hudson valley | hudson valley | hudson valley | medford-ashland | erie |
| **price** | 6000 | 11900 | 21000 | 1500 | 4900 | 1600 | 1000 | 15995 | 5000 | 3000 |
| **year** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **manufacturer** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **model** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **condition** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **cylinders** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **fuel** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **odometer** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **title_status** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **transmission** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **VIN** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **drive** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **size** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **type** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **paint_color** | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| **state** | az | ar | fl | ma | nc | ny | ny | ny | or | pa |

```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 426880 entries, 0 to 426879
Data columns (total 18 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   id            426880 non-null  int64
 1   region        426880 non-null  object
 2   price         426880 non-null  int64
 3   year          425675 non-null  float64
 4   manufacturer  409234 non-null  object
 5   model         421603 non-null  object
 6   condition     252776 non-null  object
 7   cylinders     249202 non-null  object
 8   fuel          423867 non-null  object
 9   odometer      422480 non-null  float64
 10  title_status  418638 non-null  object
 11  transmission  424324 non-null  object
 12  VIN           265838 non-null  object
 13  drive         296313 non-null  object
 14  size          120519 non-null  object
 15  type          334022 non-null  object
 16  paint_color   296677 non-null  object
 17  state         426880 non-null  object
dtypes: float64(2), int64(2), object(14)
memory usage: 58.6+ MB
```

# Data cleaning:

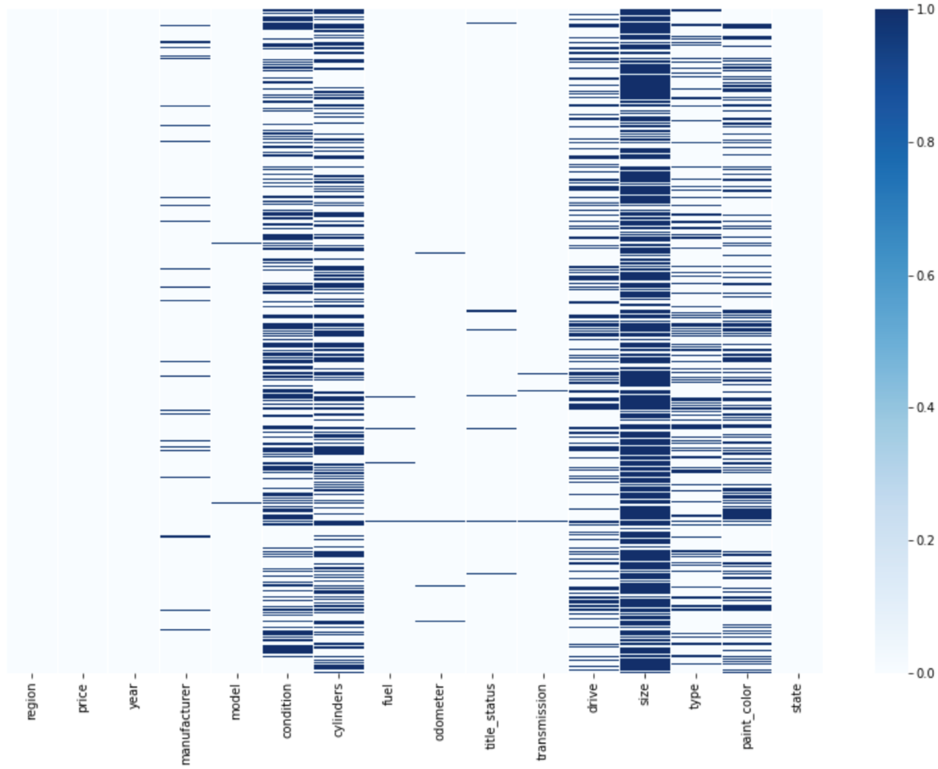
The first step for data cleaning was to remove unnecessary features. As a next step, it was

[ ]:

| | null | percent |
|---|---|---|
| size | 30636100 | 71.767 |
| cylinders | 17767800 | 41.622 |
| condition | 17410400 | 40.785 |
| drive | 13056700 | 30.586 |
| paint_color | 13020300 | 30.501 |
| type | 9285800 | 21.753 |
| manufacturer | 1764600 | 4.134 |
| title_status | 824200 | 1.931 |
| model | 527700 | 1.236 |
| odometer | 440000 | 1.031 |
| fuel | 301300 | 0.706 |
| transmission | 255600 | 0.599 |
| year | 120500 | 0.282 |
| region | 0 | 0.000 |
| price | 0 | 0.000 |
| state | 0 | 0.000 |

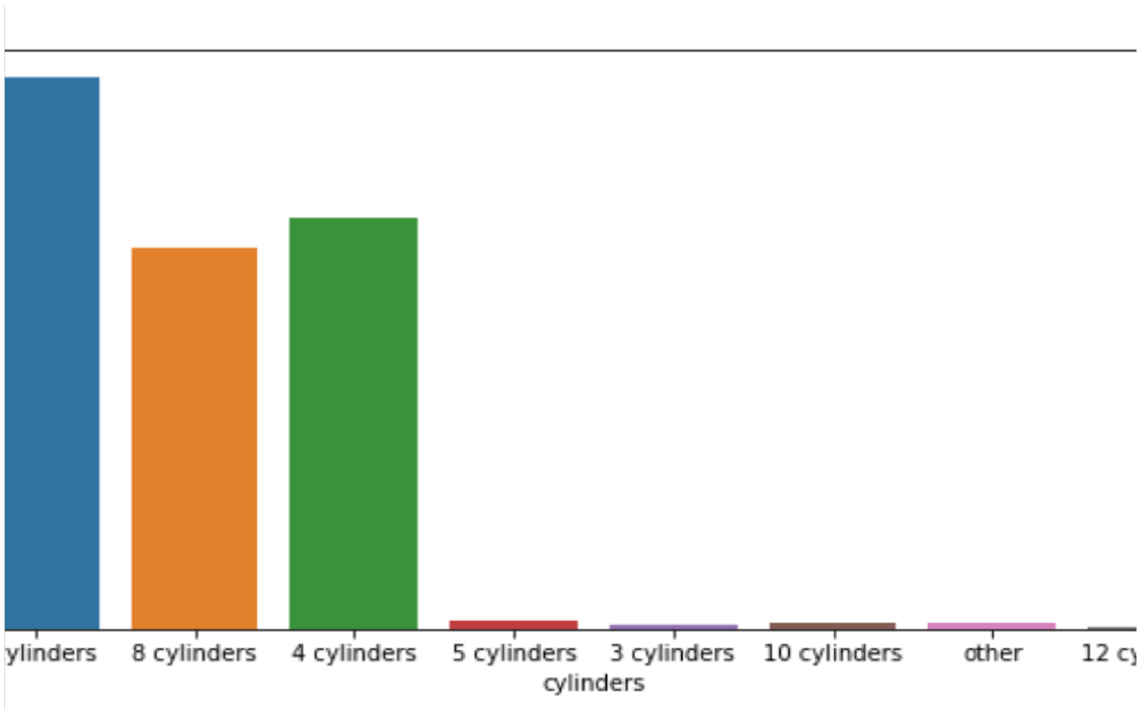
investigated the number of null points and percentage of invalid data points for that feature.

```
plt.figure(figsize=(15,10))
sns.heatmap(df1.isna(),yticklabels=False,cbar=True, cmap='Blues')
#sns.heatmap(df2.isna(), yticklabels=False, cbar=True, cmap='Accent')
plt.show()
```

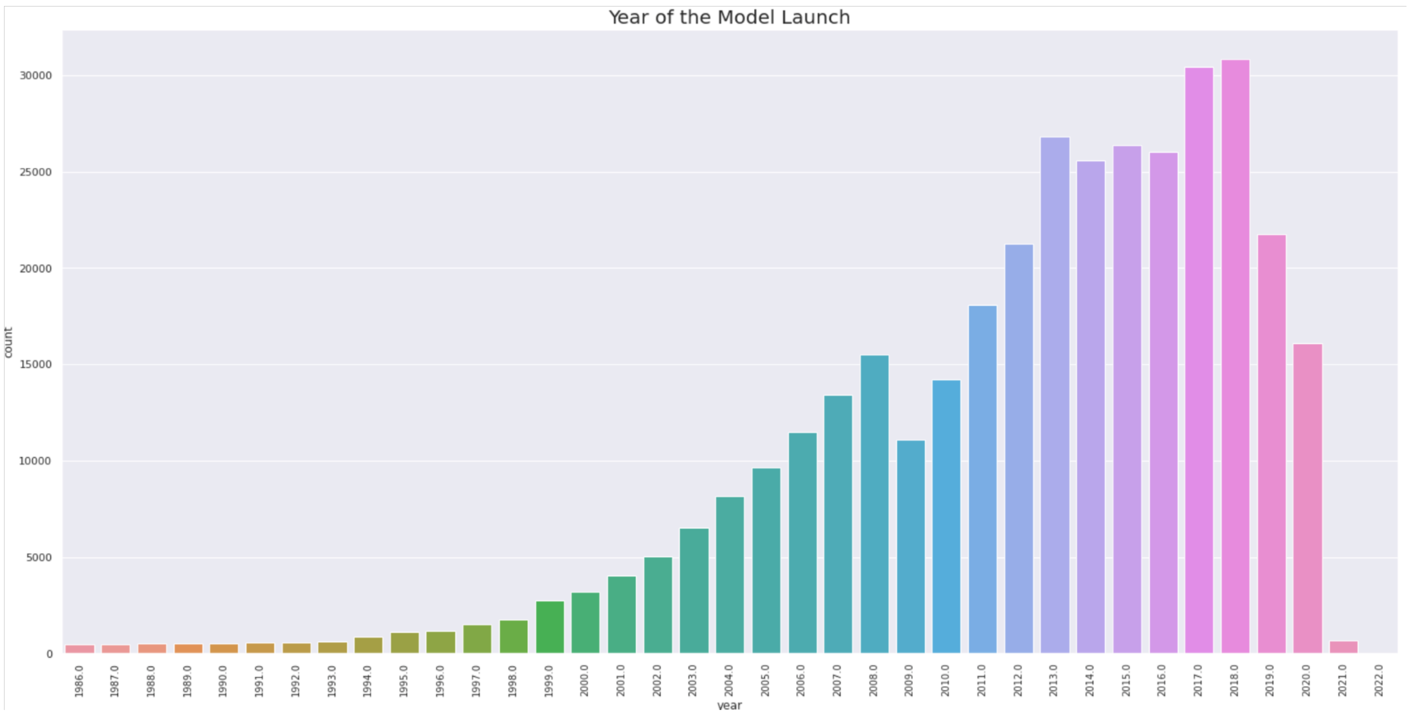


# The Exploratory Data Analysis (EDA)

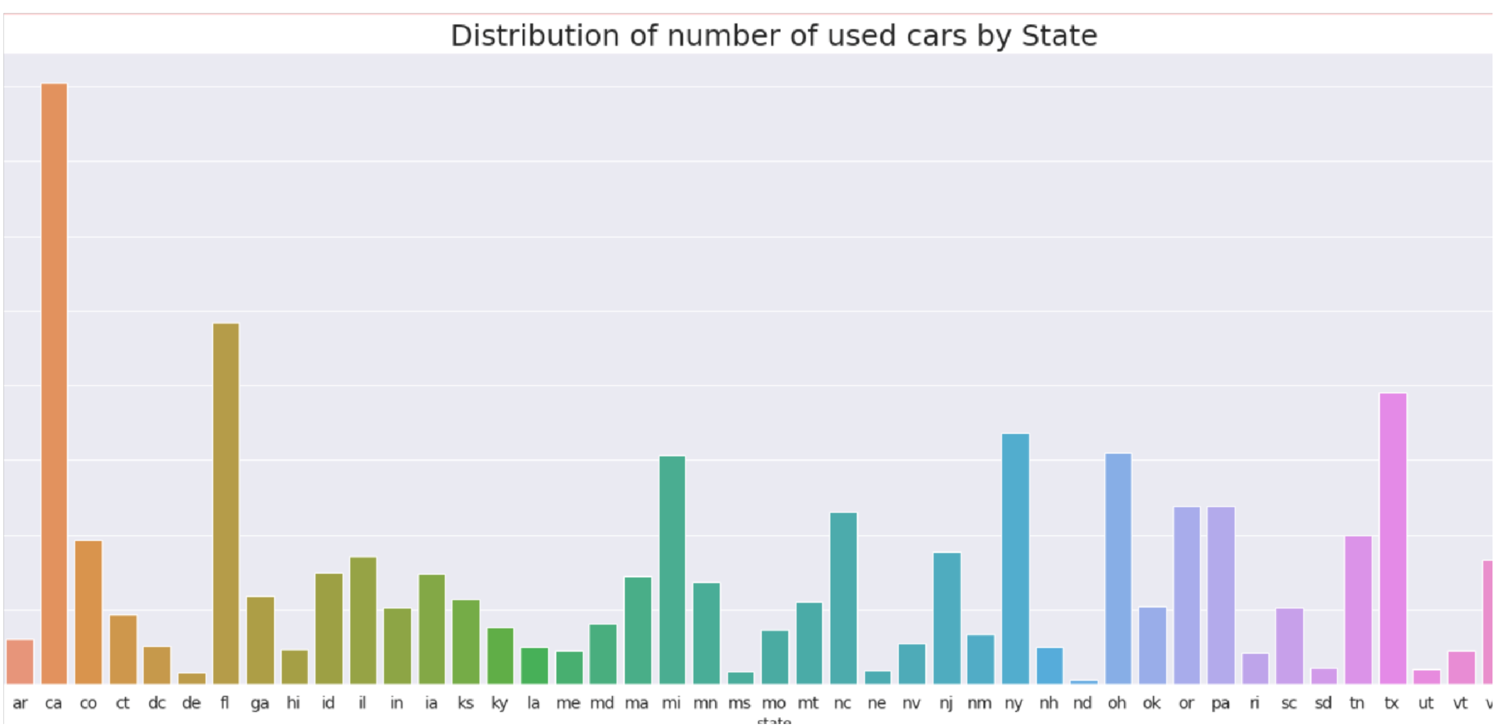
During this process, we explore the data. Our goal is to look at the different combinations of features with the help of visual graphs. This will help us to understand our data better and give us some clues about a pattern in data. Here are the valuable insights:



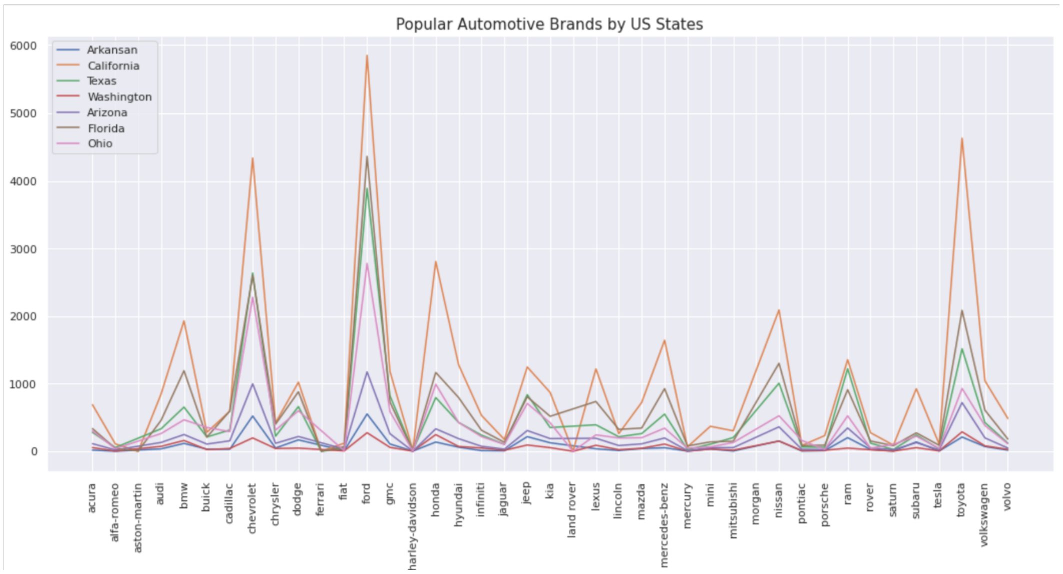

The most popular cars are 4, 6, and 8 Cylinders




Model Launch Year




Cars by state - California is a car state with the most vehicles in the US




Popular car brands in the US States

# Machine Learning Models

At this step, I applied machine learning models as a framework for data analysis. The dataset is supervised data which refers to fitting a model of dependent variables to the independent variables to accurately predict the dependent variable for future observations or understand the relationship between the variables

In this section, these machine learning models were be applied in order:

- Linear Regression
- Ridge Regression
- Lasso
- K-Nearest Neighbor

# Preprocessing the data

**Label Encoding.** In the dataset, there are 18 predictors. Two of them are numerical variables, while the rest of them are categorical. To apply machine learning models, we need the numeric representation of the features. Therefore, all non-numeric features were transformed into numerical form.

Here are the rest of the steps I took:

1. **Train the data.** In this process, 20% of the data was split into the test data, and 80% was taken as train data.
2. **Scaling the Data.** While exploring the data during the previous exploration, the data is not normally distributed. Without scaling, the machine learning models will try to disregard coefficients of features that have low values because their impact will be so small compared to the enormous value features.

# Linear Regression

Before applying ridge and lasso, examine linear regression results for an initial test. As seen in the attached notebook, the performance of the linear regression was not outstanding.

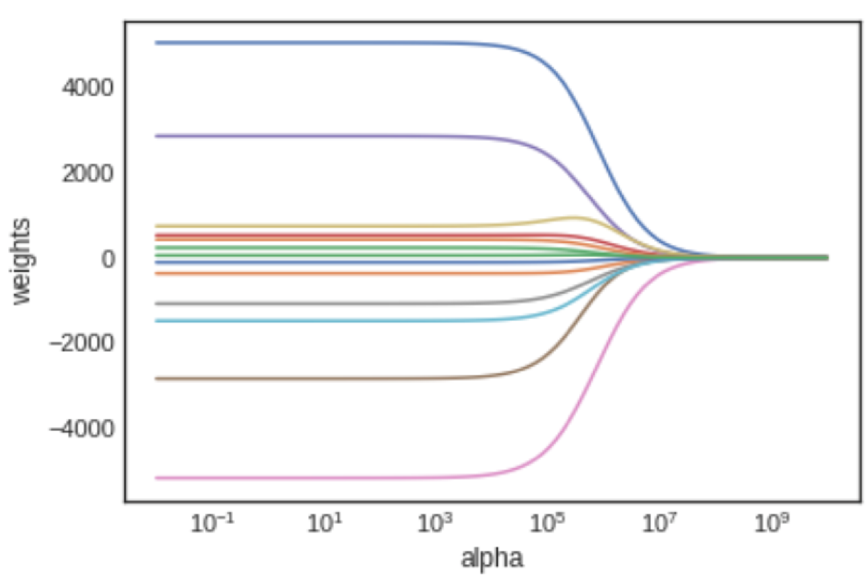
```
-------------Linear Regression-------------
RMSE = 9895.71
Accuracy =  51.083507641643976 %
-------------Ridge Regression--------------
RMSE = 9895.71
```

Table with RMSE

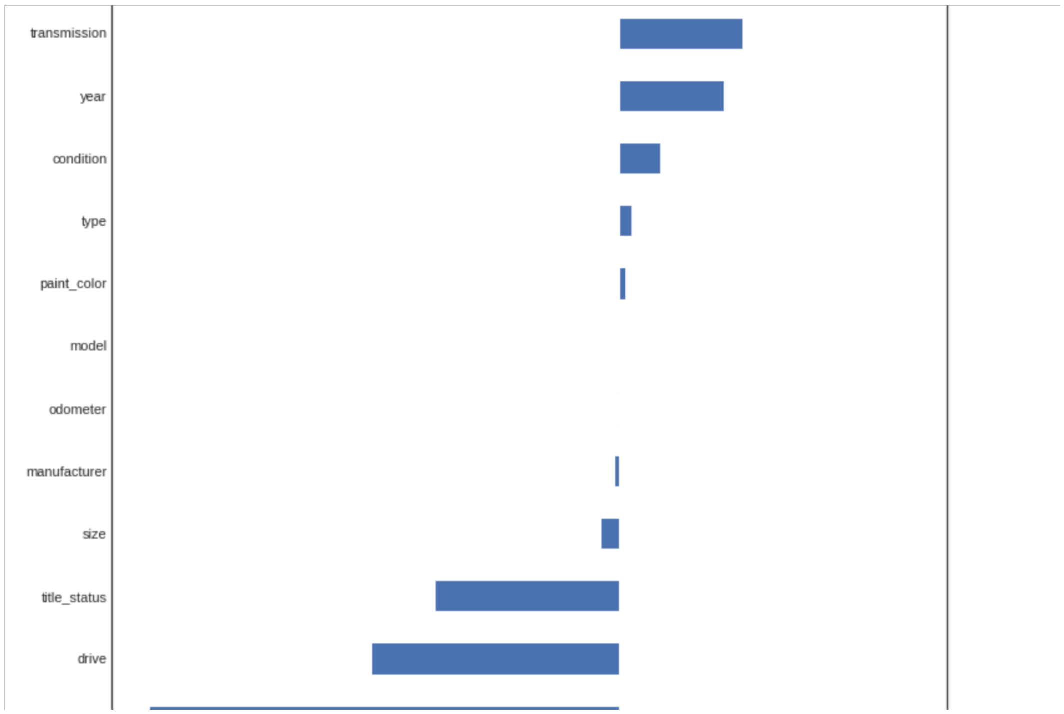
# Ridge Regression

Ordinary least square (OLS) gives unbiased regression coefficients (maximum likelihood estimates "as observed in the dataset"). Ridge regression and lasso allow to regularize ("shrink") coefficients.

```
Text(0, 0.5, 'weights')
```



To find the best alpha value in ridge regression, I used cross-validation. The results are:

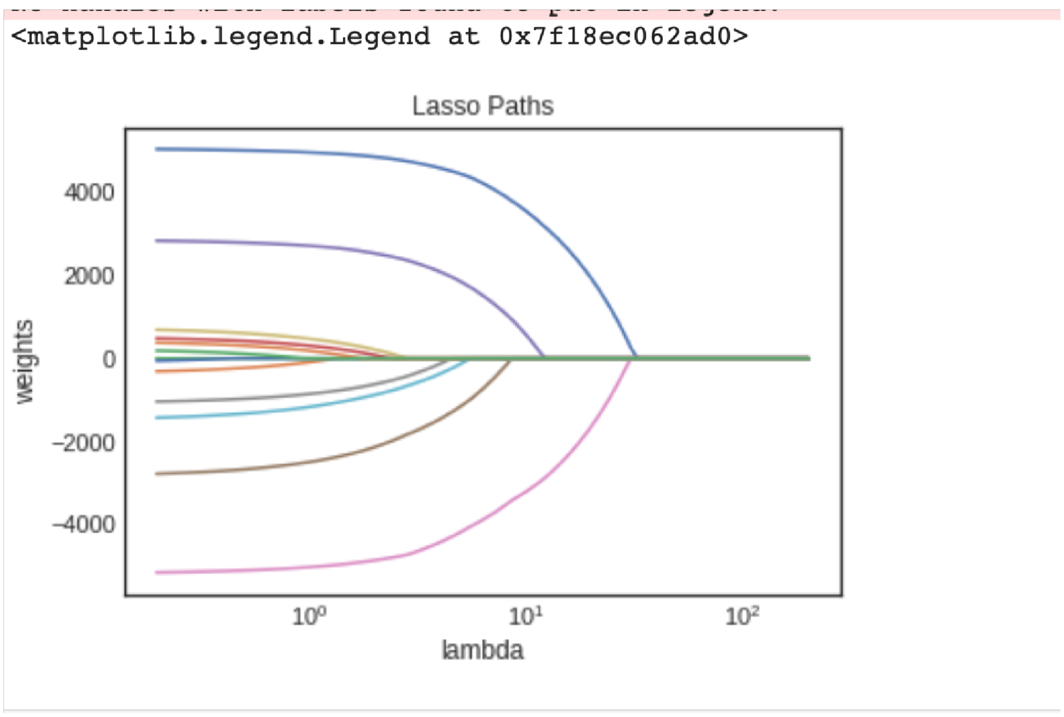



Features

Compared to OLS, the performance of the ridge is almost identical. Ridge regression suggests that these six variables are the most important: year, odometer, fuel, cylinders, title status, and drive.

# Lasso

Ridge shrinks the coefficients of the variables but does not make them zero. This may be good for some instances, but it can create a challenge in model interpretation in settings where the number of variables is quite large. For this dataset, the number of variables is not large, so there is no serious need for a lasso model. However, looking at the lasso can give us another perspective. There is no harm to applying lasso to the dataset other than investing time and effort.




Caption

```
training score for alpha =  0.5132435195918412
test score for alpha =  0.5101448807790192
number of features used: for alpha =  12
```

Training and Test Scores


transmission
year
condition
type
paint_color
model
odometer
manufacturer
size
title_status
drive

Features

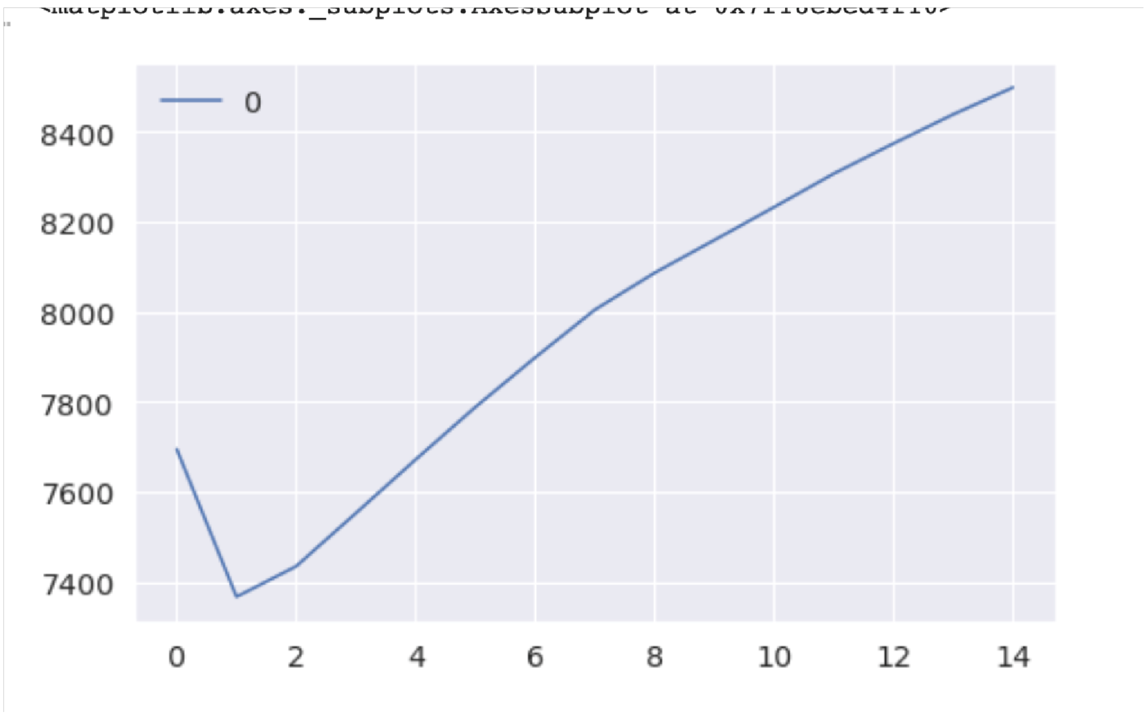
# K-nearest Neighbors (KNN)

KNN-classifier can be used when your data set is small enough so that KNN-Classifier completes running in a shorter time. The KNN algorithm can compete with the most accurate models because it makes highly accurate predictions. Therefore, we can use the KNN algorithm for applications that require an excellent prediction but do not require a human-readable model. The quality of the predictions depends on the distance measured.

Evaluation of the RMSE values are shown below:

```
RMSE value for k=  1 is: 7695.424317147372
RMSE value for k=  2 is: 7367.400813939839
RMSE value for k=  3 is: 7435.2718809156895
RMSE value for k=  4 is: 7553.182741915963
RMSE value for k=  5 is: 7672.244714175627
RMSE value for k=  6 is: 7788.745904018688
RMSE value for k=  7 is: 7898.851488508709
RMSE value for k=  8 is: 8005.034626749569
RMSE value for k=  9 is: 8086.951550524175
RMSE value for k=  10 is: 8159.6288128156775
RMSE value for k=  11 is: 8233.16978371386
RMSE value for k=  12 is: 8307.570671954567
RMSE value for k=  13 is: 8374.549649999415
RMSE value for k=  14 is: 8439.268463843975
RMSE value for k=  15 is: 8498.914467451295
```

RMSE for K Values




RMSE value is at its lowest when k is one.

# Conclusion

Using different models, our goal was to get different perspectives and eventually compare their performance. We predicted the price of used cars using a dataset with ten predictors and approximately 359410 rows after cleaning.

We uncovered great insights and explored the data, as seen in the graphs above of the data visualizations during the exploratory data analysis. Finally, we applied predictive models to predict the price of used cars in order (linear regression, ridge regression, lasso, and KNN). However, this was a relatively small dataset for making a factual inference because the number of observations was only 359410. Gathering more data will yield more robust predictions. Also, we can improve the data cleaning process with the help of more technical information.

As a suggestion for further studies, I will use one hot encoder method for better analysis while preprocessing data instead of using a label encoder. Thus, all non-numeric features can be converted to nominal data instead of ordinal data.